

WHAT IS CLAIMED IS:

1. A data-driven processor provided with a memory with n (where n is any integer of at least 2) regions each having a k-bit length (where k is any positive integer), comprising an operation processing portion for performing an operation on data stored in said n regions of said memory (hereinafter referred to as memory data), said operation processing portion including:

prescribed operation performing means for (i) performing a prescribed operation on said memory data and said k-bit length data stored in a data field of an applied data packet in accordance with a prescribed operation instruction, (ii) dividing a result of said prescribed operation into a plurality of said k-bit length data, and (iii) outputting a plurality of said data packets storing said plurality of k-bit length data obtained by the division in said data fields;

data accumulating means for (i) receiving said plurality of data packets output from said prescribed operation performing means, and for each of received plurality of data packets, ⁽ⁱ⁾~~(ii-1)~~ accumulating said k-bit length data in said data field of the received data packet on the memory data in said region at a prescribed address of said memory, ⁽ⁱⁱⁱ⁾~~(ii-2)~~ storing the accumulation result excluding overflowing data in that region, and ^(iv)~~(ii-3)~~ outputting the received data packet storing said overflowing data in said data field; and

overflowing data accumulating means for (i) receiving said data packet storing said overflowing data in said data field, (ii) accumulating said overflowing data in said data field of the received data packet onto the memory data in the region at an upper prescribed address different from said prescribed address in said memory, (iii) storing the accumulation result excluding said overflowing data in the region, and (iv) outputting the received data packet storing said overflowing data in said data field, wherein

the accumulation of said overflowing data by said overflowing data accumulating means is repeated as long as the accumulation causes said

overflowing data in the region at said prescribed address.

2. The data-driven processor according to claim 1, wherein
said operation processing portion further includes overflow
determining means for determining if said overflowing data occurs in the
region at said prescribed address by the accumulation, and the
5 accumulation is performed on said overflowing data by said overflowing
data accumulating means in response to a determination by said overflow
determining means that said overflowing data occurs in the region.

3. The data-driven processor according to claim 1, wherein
when two multiple-precision data each having a m (where m is any
integer satisfying $n * k \geq m$)-bit length are subjected to said prescribed
operation, data obtained by dividing one of said multiple-precision data by
every said k -bit length are stored in the n regions of said memory as said
5 memory data, respectively, and data obtained by dividing the other of said
multiple-precision data by every said k -bit length are stored in said data
fields of said n data packets, respectively, which are sequentially applied to
said prescribed operation performing means.

4. The data-driven processor according to claim 1, wherein
said data packet further includes a generation field storing a
generation number for uniquely identifying said data packet, and said
prescribed address is designated based on a content of said generation field
5 of said data packet.

5. The data-driven processor according to claim 1, wherein
each of said data accumulating means and said overflowing data
accumulating means is operated in accordance with an operation instruction
for receiving ^{said applied} ~~applied~~ said data packet, accumulating a content of said data
5 field in the received data packet on the memory data in said region at said
prescribed address of said memory, storing the accumulation result
excluding overflowing data in said region, and storing said overflowing data

in said data field of the received data packet for output.

6. A data processing method in a data-driven processor provided with a memory with n (where n is any integer of at least 2) regions each having a k -bit length (where k is any positive integer) and storing data in said n regions (hereinafter referred to as memory data), comprising:

5 a prescribed operation performing step for (i) performing a prescribed operation on said memory data and the data having said k -bit length stored in a data field of an applied data packet in accordance with a prescribed operation instruction, (ii) dividing the result of said prescribed operation into a plurality of said k -bit length data, and (iii) outputting a
10 plurality of said data packets storing said plurality of k -bit length data obtained by said division in said data fields, respectively;

15 a data accumulating step for ⁽ⁱ⁾~~(i-1)~~ accumulating said k -bit length data in said data field of said data packet on memory data in said region at a prescribed address of said memory, ⁽ⁱⁱ⁾~~(i)~~ receiving said plurality of data packets output from said prescribed operation performing step, and for each of received plurality of data packets, ⁽ⁱⁱⁱ⁾~~(i-2)~~ storing the accumulation result excluding overflowing data in the region, and ^(iv)~~(i-3)~~ outputting said data packet storing said overflowing data in said data field; and

20 an overflowing data accumulating step for (i) receiving said data packet storing said overflowing data in said data field, (ii) accumulating said overflowing data in said data field of the received data packet on the memory data in said region at an upper prescribed address different from said prescribed address in said memory, (iii) storing the accumulation result excluding said overflowing data, and (iv) outputting the received data
25 packet storing said overflowing data in said data field, wherein the accumulation of said overflowing data by said overflowing data accumulating step is repeated as long as the accumulation causes said overflowing data in the region at said prescribed address.